



IBM Informix® Dynamic Server™ (IDS)
IDS Problem Determination Tutorial Series

Gateway Connectivity Problem Determination

Table of Contents

About this tutorial	3
Introduction.....	3
Setup	3
Tutorial Conventions Used	3
About the author.....	3
Section 1 Understanding The Gateway Environment.....	4
Section 1.1 Installation.....	4
Section 1.2 Directory Structure.....	4
Section 1.3 Working of the Gateway Daemon	5
Section 1.4 Supported Databases.....	6
Section 1.5 Starting the Gateway Manager.....	6
Section 2 Problem Determination.....	8
Section 2.1 Executing a Query Involving a Remote Table Hangs or Results in an error 908.....	8
Section 2.2 Errors due to missing links.	8
Section 2.3 Errors Starting Gateway Daemon.....	9
Section 2.4 Invalid Username/password error	9
Section 3 Confirming Your Gateway Environment	11
Section 3.1 Confirming That the Gateway Daemon Is Running	11
Section 3.2 Confirming User Mapping	11
Section 3.3 Confirming That the ODBC File Is Correct	11
Section 3.4 Verification of a Simple Connection Using egmdba	13
Section 3.5 Verification of sqlhosts and Services.....	13
Section 3.6 Verification of Userid Mapping.....	13
Section 3.7 Verification of ODBC Driver Availability	14
Section 3.8 Queries Taking a Long Time	14
Section 4 Collecting Diagnostic Information For Gateway Issues	15
Section 4.1 Gateway Diagnostics	15
Section 4.2 ODBC Diagnostics.....	15
Summary	16
What you should know	16
For more information.....	16

Gateway Connectivity Problem Determination

About this tutorial

Introduction

The objective of this tutorial is to give you a fundamental understanding of IDS connectivity when using a gateway. The tutorial will guide you through examples that will familiarize you with the tools and methods to determine the cause of connectivity problems in this type of environment.

This tutorial is intended for DBA's who would be using Informix Enterprise Gateway Manager with ODBC to establish connectivity between Informix and non-Informix database servers. It is assumed that the reader has an understanding of Informix Online Server, setting up connectivity between client application (based on ESQL/C or 4GL) and Informix Server. The user should also be familiar with SQL and know how to execute distributed queries using remote Informix Servers.

Setup

The examples in this tutorial are based on the Informix Gateway Manager running on Solaris and the remote database to be Oracle. Information regarding other database servers will be presented if necessary. This tutorial assumes that you have a Informix Online Server and an Oracle server.

Tutorial Conventions Used

When a tool or utility is first mentioned it will be shown in **bold** text.

All command statements and their output will be shown in a `monospaced` font.

Some examples will show specific command options which may change over time, which will always be documented in IDS documentation.

About the author

Srinath Shenoy, B.S. (Computers) is a member of Advanced Technical Support Group for IBM Informix Client products and Informix Enterprise Gateway Manager. As a development engineer in ClientSDK group, Srinath has worked on enhancing DB utilities to support Informix Universal Server, IBM-Informix front-end tools and the development of JDBC driver.

Section 1 Understanding The Gateway Environment

Section 1.1 Installation

You will need to login as root to install gateway. Here are the steps:

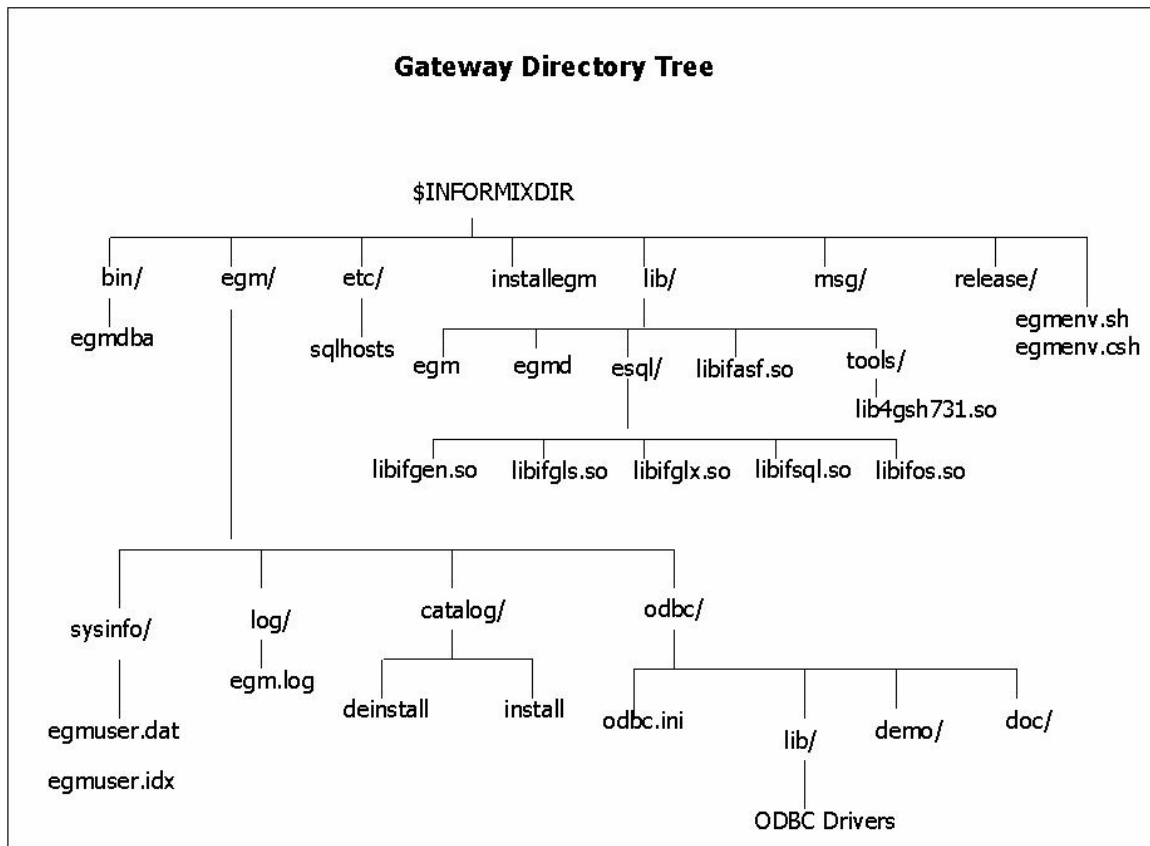
1. Copy the product distribution media to the directory where you want to install the gateway. The directory where gateway will be installed should not be an NFS mounted directory.
2. Set the environment variable INFORMIXDIR to the directory where gateway will be installed.
3. Run the script installegm as user root. This will install the product.
4. While installing the product you will be prompted for a serial number and a key.

Section 1.2 Directory Structure

The directory structure for Informix gateway is similar to **IDS** Online Server. The following directories are created on installing the gateway:

- bin - egmdba, bcheckegm,
- lib - the gateway daemon and various libraries required by the gateway
- etc - termcap, sqlhosts
- msg - message files
- release - the release notes
- gls - codeset conversion files.

In addition you will find a directory called egm. This directory has information which is specifically required by the gateway manager. This includes the ODBC libraries and scripts to populate catalog tables on remote databases.

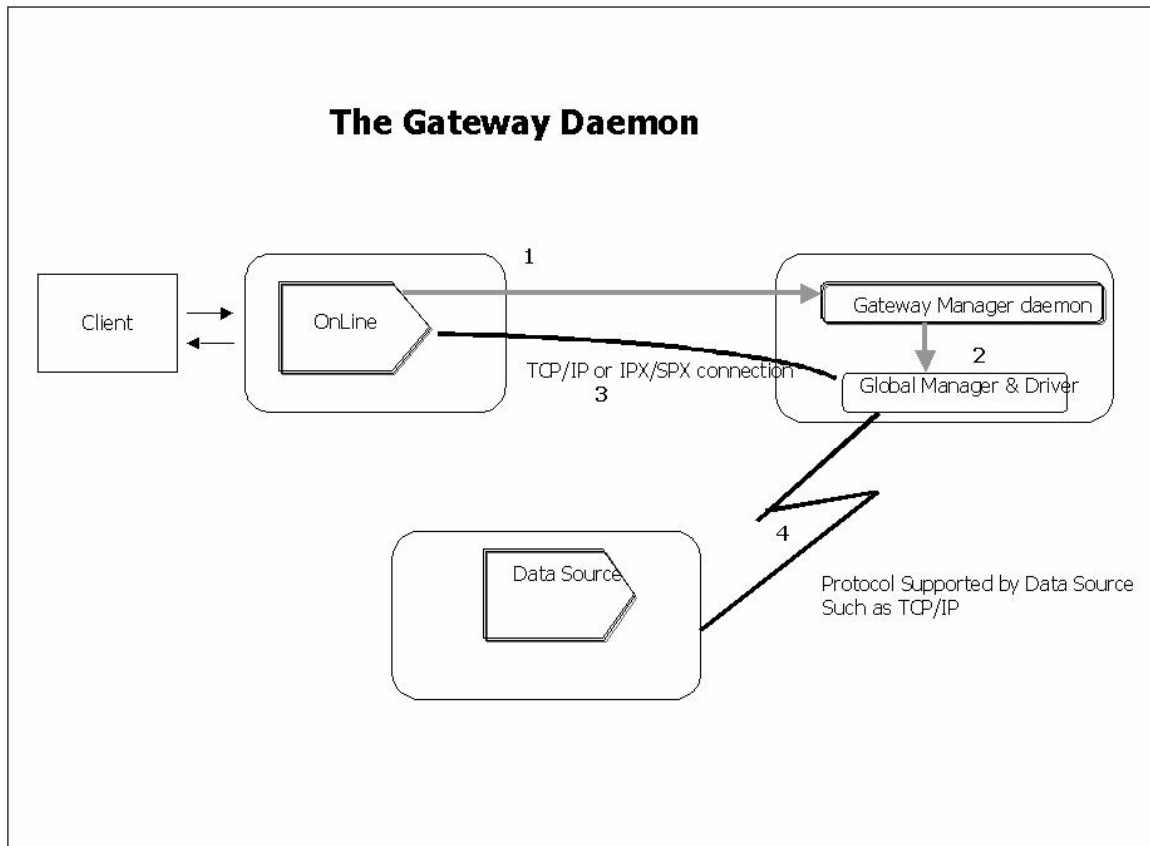


Section 1.3 Working of the Gateway Daemon

When a client application connects to IDS and information from remote database is requested, IDS will make a connection request to the gateway manager daemon. The information required to make the connection is read from `sqlhosts` file on the client side.

The Gateway manager will start a new gateway manager process, connect the Online server to the newly spawned process and detach itself from the connection to the Online server. The newly spawned gateway manager process then talks to the remote database based on the ODBC configuration.

The following figure shows the working of the gateway process:



Section 1.4 Supported Databases

Currently the following databases are supported:

- DB2
- Oracle
- Sybase
- SQL Server
- dBASE

Section 1.5 Starting the Gateway Manager

There are 2 files `egmenv.csh` and `egmenv.sh` under `INFORMIXDIR`. Depending on the shell you are using, you can modify these files to customize your environment. The most important environment variables here are `INFORMIXDIR`, `INFORMIXSERVER`, `PATH`, `INFORMIXSQLHOSTS`, `ODBCINI` and `LD_LIBRARY_PATH` (`SHLIB_PATH` on HP)

To start the gateway execute the following as root:

```
$INFORMIXDIR/lib/egmd <Informix server name> -s egm -l <log file>
```

where <Informix server name> is the name of the gateway server configured in the sqlhosts file and <log file> is the log file for the gateway daemon.

Section 2 Problem Determination

Section 2.1 Executing a Query Involving a Remote Table Hangs or Results in an error 908.

Let's assume that you have configured gateway server to talk to a remote database. Consider the query listed below, which is execute in dbaccess

```
select * from wireora@egm_dan:'SCOTT'.tab;
```

If you see the error as shown below:

```
select * from wireora@egm_dan:'SCOTT'.tab
#                                     ^
#  908: Attempt to connect to database server (egm_dan, conerr=-
908, oserr=13) failed.
#
```

This indicates that your gateway server is not running. You may have configured your gateway server but may not have started it. To simulate this error, bring down your gateway server and run a query which accesses a remote table. If you see this error, check if the machine has been rebooted or someone has killed the gateway daemon. If you need to start the gateway everytime the machine is rebooted, check with your system administrator to make an entry in /etc/init.d/inittab.

Section 2.2 Errors due to missing links.

Since the gateway daemon runs as root, it will look only at /usr/lib for all the shared libraries. The gateway daemon relies on ODBC libraries to establish connectivity to remote databases. The EGM install script in Version 7.31.UD1 is designed to link all the ODBC drivers to /usr/lib. In case of prior releases, if the user has not linked libraries in /usr/lib one of the error messages is as shown below:

```
select * from wireora@egm_dan:'SCOTT'.tab
#                                     ^
#  908: Attempt to connect to database server (egm_dan, conerr=-27001, oserr=8)
failed.
#
```

In this case, check the gateway manager log file. This is the file that you specify when you start the gateway daemon. If you see a message

```
2003-05-08 16:49:16.486261 srvinfx dan shenoy -nwireorald.so.1:
/extra1/gateways/731uc2/lib/egm: fatal: libodbc.so: open failed: No such file or
directory
```

or

```
2003-05-08 16:57:10.230234 srvinfx dan shenoy -nwireora ld.so.1:
/extra1/gateways/731uc2/lib/egm: fatal: libodbcinst.so: open failed: No such file
or directory
```

then, make sure that you have links to libodbc.so and libodbcinst.so in /usr/lib.

The following error is a generic error you will see in case some of the links are missing in /usr/lib.

```
select * from wireora@egm_dan:'SCOTT'.tab
#                                     ^
#29081: ODBC Error ([80 S1000 :00.00.0000]).
#
```

The section "Confirming your gateway environment" explains how to identify the libraries to be linked in /usr/lib.

Section 2.3 Errors Starting Gateway Daemon

In case the gateway is already running you will see the error shown below:

```
2003-05-08 17:26:27.395852 daemon err = -25572: oserr
= 22: errstr = : Network driver cannot bind a name to
the port. System error = 22.
```

In case you specify an invalid protocol in sqlhost file, you will see the error listed below:

```
2003-05-08 17:31:10.201591 daemon err = 25507: oserr =
0: errstr = : The specified service name or protocol
is unknown.
```

Please remember to check your log file in case you have used the "-l" option while starting the gateway. These messages are not printed on the screen but in the log file in case you start the gateway with the logging option.

Section 2.4 Invalid Username/password error

If you see the following error:

```
select * from wireora@egm_dan:'SCOTT'.tab
#                                     ^
#29080: Target DBMS Error ([1017 28000 :00.00.0000]
ORA-01017: invalid username/password; logon de).
#
```

Check the log file to see the detailed error message. The message would be as shown below.

```
Error/warning message received:
```

```
"[1017 28000 :00.00.0000] [DATADIRECT][ODBC Oracle  
Wire Protocol driver][Oralle]ORA-01017: invalid  
username/password; logon denied"
```

This is an indication that the user running the query on IDS has not been mapped to a valid user on the remote database server. Every Informix user who wishes to communicate with the remote server needs to be mapped to a valid user on the remote database server for gateway.

Section 3 Confirming Your Gateway Environment

Section 3.1 Confirming That the Gateway Daemon Is Running

To confirm that the gateway daemon is running use the ps command. Here is a sample output:

```
dan:u/shenoy ps -aef | grep egm
root  9715      1  0 17:37:06 pts/15    0:00 lib/egmd egm_dan -s
egm -l /tmp/egm.log
shenoy 10458    714  0 12:37:48 pts/22    0:00 grep egm
```

The egm daemon will be running as root. If the gateway server daemon is not running, you can start it as explained in previous section.

Section 3.2 Confirming User Mapping

To map user's you will need to use egmdba. egmdba is a tool that lets you map users, test connection to the data source and install Informix style catalogs on the remote database. To install catalogs or add/delete/update users other yourself, you will need to login as user 'informix'. This tool is located in \$INFORMIXDIR/bin. Make sure that this directory is located in your path.

1. Login as informix and set your gateway environment.
2. Now start egmdba. This will show a menu.
3. Select the "User" option. This will take give you three options, to add a users and find existing users.
4. Select the find option, to check if the user connecting to remote database is mapped.
5. Enter the user login in the "Unix Userid" filed and press "Escape". This will list all the user mappings for the user you just entered.
6. The Next or Previous option will let you scroll through the selected records.

When users are defined for the first time, the directory \$INFORMIXDIR/egm/sysinfo is created. The users that are mapped are stored in egmuser.dat and egmuser.idx. Never modify these files! If the user does not exist for a particular datasources, create a new user mapping. You will be prompted for the password twice.

Once you have added the user, you can verify that the user has been added using the Find option explained above.

Section 3.3 Confirming That the ODBC File Is Correct

Gateway relies on ODBC to establish connectivity to the remote database. Along with the product, is an executable called demoodbc which talks to the remote database.

- **demoodbc** is located under \$INFORMIXDIR/egm/odbc/demo.

- **demoodbc** uses your odbc.ini entries to talk to remote database and is independent of the gateway.

Before you run demoodbc, run the sql script emporacle.sql against your Oracle server. This will create a table emp and insert some data into it. What demoodbc does is print this data. In the example shown below, demoodbc uses a datasource called wireora and user/password combination of scott/tiger.

```
$INFORMIXDIR/egm/odbc/demo/demoodbc -uid scott -pwd tiger  
wireora/extral/gateways/731uc2/egm/odbc/demo/demoodbc MERANT, Inc.
```

ODBC Sample Application:

/extral/gateways/731uc2/egm/odbc/demo/demoodbc: will connect to data source 'wireora' as user 'scott/tiger'.

First Name	Last Name	Salary	Dept
-----	-----	-----	----
Tyler	Bennett	1977-06-01	00:00D101 32000.00
John	Rappl	1987-07-15	00:00D050 47000.00

In case you have not created the table EMP, you get the error as shown below:

```
$INFORMIXDIR/egm/odbc/demo/demoodbc -uid scott -pwd tiger  
wireora/extral/gateways/731uc2/egm/odbc/demo/demoodbc DATADIRECT  
TECHNOLOGIES, INC.
```

ODBC Sample Application.

```
/extral/gateways/731uc2/egm/odbc/demo/demoodbc: will connect to data  
source 'wireora' as user 'scott/tiger'.  
.....SQLExecute has Failed. RC=-1  
SQLSTATE = 42S02  
NATIVE ERROR = 942  
MSG = [DATADIRECT][ODBC Oracle Wire Protocol driver][Oracle]ORA-00942:  
table or view does not exist
```

In case demoodbc is not able to connect to the remote server, you may see a message as shown below.

```
SQLConnect: Retrying Connect.  
SQLConnect: Failed...  
SQLSTATE = HY000  
NATIVE ERROR = -1  
MSG = [DATADIRECT][ODBC Oracle Wire Protocol  
driver][Oracle]ORA-12203: unable to connect to  
destination
```

This indicates that there is a problem with your datasource. Things to verify in the datasource when demoodbc fails are:

- Is the machine name right? Try using ip-address
- Is the port number correct?
- Am I using the correct library?
Note that for Oracle there are 2 drivers. One is wired protocol driver and other is non-wired protocol driver. If you are using non-wired protocol driver you need Oracle client libraries on the machine where gateway is running.
- Is demoodbc reading the correct odbc.ini file?
By default the odbc driver will look in the user's home directory for an odbc.ini file. This can be overwritten by environment variable ODBCINI. In gateway environment file, ODBCINI file is set to \$INFORMIXDIR/egm/odbc/odbc.ini. Confirm that the correct file is being looked up.

Section 3.4 Verification of a Simple Connection Using egmdba

The egmdba has a test-connect option. You can use this option to verify that your gateway can talk to the remote server. Once you have verified that demoodbc works, you can try this option in egmdba. It will prompt you for the datasource name and the gateway server name. If connection is successfully established, it will take you to the select-count-test screen.

Section 3.5 Verification of sqlhosts and Services

Verify that the sqlhost entry is the correct format. Confirm that there is an entry in the sqlhosts file for the gateway server. If you are using a service name, make sure that the service name is defined in /etc/services and that the port number used is correct. When you connect to the gateway server, the datasources defined will be shown as list of databases. Confirm that the protocol used is right for that platform. For example, for Solaris you will use ontltcp.

Section 3.6 Verification of Userid Mapping

Every time a connection is established an entry is made in egm.log under \$INFORMIXDIR/egm/log. If a connection is successful, you will see an entry as shown below:

```
2003-05-08 17:19:22.271564 PID= 9651 Client User shenoy (Gateway
Process User shenoy) connected to Data Source wireora as user
scott
```

This will give you an indication as to what user mapping has been done. In the above example user shenoy has been mapped to user scott for datasource wireora. In case there is a problem with user/password combination, you will see the following error message:

```
2003-05-09 15:21:44.209932 PID=10572 Client User shenoy (Gateway
Process User shenoy) failed to connect to Data Source wireora as
```

```
user scott. SQLCODE=-29080, Message Token(s)='[1017 28000
:00.00.0000] ORA-01017: invalid username/password; logon de'.
```

There was no user mapping for user shenoy and user shenoy did not exist on the Oracle side. In dbaccess the error seen is:

```
select * from wireora@egm_dan:'SCOTT'.tab
#                                     ^
#29080: Target DBMS Error ([1017 28000 :00.00.0000] ORA-01017:
invalid username/password; logon de).
#
```

The entry in log file under \$INFORMIXDIR/egm/log is:

```
2003-05-09 15:23:56.410846 PID=10577 Client User shenoy (Gateway
Process User shenoy) failed to connect to Data Source wireora as
user shenoy. SQLCODE=-29080, Message Token(s)='[1017 28000
:00.00.0000] ORA-01017: invalid username/password;
logon de'.
```

The egm.log under \$INFORMIXDIR/egm/log logs connection and disconnection activity.

Section 3.7 Verification of ODBC Driver Availability

Drivers for all UNIX platforms have the same file name. The extensions are differentiated for Solaris, HP-UX, AIX and Linux as follows:

```
Solaris: filename.so
HP-UX:   filename.sl
AIX:     filename.so
```

Section 3.8 Queries Taking a Long Time

Check if you have populated the system catalogs. If the Informix style system catalogs have not been populated, populate the system catalogs using the Add-tables option. If the system catalog tables have been populated, use the Refresh-tables to update the already existing information in system catalogs. On the IDS side, do a set query on and check what optimizer plan is being generated. The query plan will give you an idea of what values the Informix optimizer gets from the gateway. The gateway gets these values from the system catalogs installed on the remote server or via ODBC APIs depending on the setting of environment variable GWCATALOG.

Check if these values received by the optimizer are correct. Also check what the environment variable GWCATALOG is set to.

If you have not installed Informix Catalogs on the remote server set GWCATALOG to PROC=3, STAT=0.

Section 4 Collecting Diagnostic Information For Gateway Issues

Section 4.1 Gateway Diagnostics

To turn on gateway debugging you can set the GWDEBUG variable on. The possible values for this environment are 121, 122 and 123. You can set all the three values seperated by "," as shown below. In C-Shell this is will be done as:

```
setenv GWDEBUG 121,122,123
```

This will create a egmt.euidpid, for example egmt.shenoy10452, where shenoy represents the user id and 10452 is the process id. You will need to restart the gateway server. To avoid restarting the gateway server, put the entry in .informix file in user's home directory as shown below:

```
GWDEBUG      121,122,123
```

Do not turn on GWDEBUG in production environment. This will slow down the performance significantly.

Section 4.2 ODBC Diagnostics

The ODBC driver allows tracing. To turn on ODBC tracing you will need to modify your ODBCINI file. Set the value of variable Trace to 1 and Tracefile to a filename where the odbc diagnostic messages need to be written to. Here is how the entry looks like:

```
[ ODBC ]
Trace=1
TraceFile=/tmp/odbctrace.out
TraceDll=/extral/gateways/731uc1/egm/odbc/lib/odbc trac.so
InstallDir=/extral/gateways/731uc1
ConversionTableLocation=/extral/gateways/731uc1/egm/odbc/tables
```

Summary

What you should know

By now you should have an idea of how the Informix Gateway Server installs and works. You should also be able to successfully configure the Gateway Server and establish connectivity between IBM-Informix and non-IBM-Informix database servers.

This tutorial also has given you an idea of the common errors that are seen while configuring the Gateway, and how to collect diagnostic information for the Gateway in case you need to get in touch with IBM-Informix Technical Support.

For more information

For more information, refer the IBM- Informix Enterprise Gateway Manager User Manual.